Appl. No.: 10/699,024 Filing Date: October 31, 2003

IN THE CLAIMS

Please amend the claims as follows:

1-2. (canceled)

- 3. (previously presented) The method of claim 14, wherein storing variations is performed using a pointer.
- 4. (currently amended) The method of claim 14, further comprising, before presenting the reconstructed represented sequence to a user: identifying, replets that can be used to represent multiple subsequences.
- 5. (currently amended) The method of claim 14, further comprising, before presenting the reconstructed <u>represented</u> sequence to a user: segmenting the matching subsequences into multiple parts to account for location-specific variations of the matching subsequences in the sequence data.
- 6. (currently amended) The method of claim 14, further comprising, before presenting the reconstructed <u>represented</u> sequence to a user:

storing replet information in a replet-information table using a pointer, so that equivalent replet sequences occupy single storage space.

7-9. (canceled)

10. (currently amended) The method of claim 14, further comprising, before presenting the reconstructed <u>represented</u> sequence to a user:

storing multiple views of the sequence data at multiple levels of abstraction.

11-13. (canceled)

Appl. No.: 10/699,024 Filing Date: October 31, 2003

14. (currently amended) A computer <u>system</u>-implemented method for storing and presenting sequence data, comprising:

- i) specifying a set of one or more initial replets for analysis by a computer system;
- <u>ii)</u> for each <u>initial</u> replet in the set, comparing <u>each an initial</u> replet <u>by the computer system</u> to a sequence to <u>for determininge</u> <u>by the computer system</u> a subsequence of the sequence that matches <u>each the initial</u> replet, if any;
- <u>iii)</u> generating and storing entries of match-set data structures by the computer system responsive to the comparing, one match-set data structure for each replet, a match-set data structure comprising a sequence identification to identify a <u>sub</u>sequence where a match occurs and offset information to determine a position within the sequence where the matching subsequence of the sequence is located, <u>wherein the offset information comprises a first and second position parameter, the first position parameter denoting a location in the sequence and the second position parameter denoting an offset from the location;</u>
- , and wherein redundant match-set data structures corresponding to subsequences being matched by more than one replet are not generated;
- iv) storing the generated entries of the match-set data structures in a computer readable memory by the computer system;

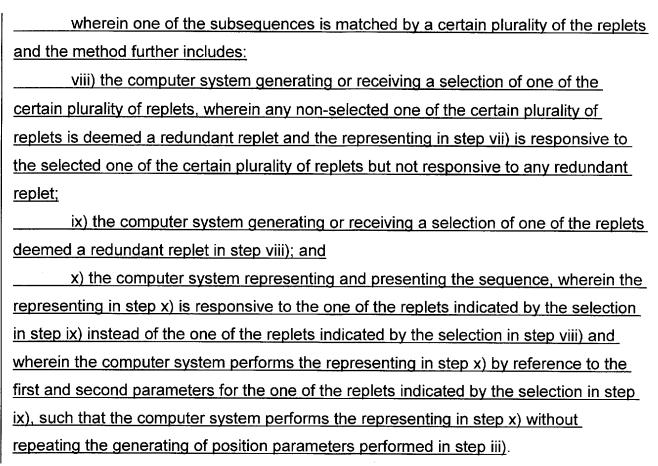
storing one or more variations for each of one or more matching subsequences, wherein a variation comprises a character in a subsequence that corresponds to a "don't care" character in a replet that matches the subsequence;

- v) deleting by the computer system each matching subsequence from the sequence where it is found;
- <u>vi)</u> concatenating by the computer system, in order, unmatched regions of sequences that remain after deleting each matching subsequence to form and store a backbone sequence;
- <u>vii) reconstructing the computer system representing the sequence a sequence</u>

 from responsive to the stored backbone sequence and at least a portion of the stored match-set data, the stored variations, and the stored backbone sequence; and presenting the reconstructed represented sequence to a user of the computer system;

Appl. No.: 10/699,024

Filing Date: October 31, 2003



- 15. (currently amended) The method of claim 14, wherein, prior to presenting the reconstructed represented sequence to the user, variations are stored in a list data structure comprising a variation identification.
- 16. (previously presented) The method of claim 15, wherein the list data structure comprises a subsequence character that matches a "don't care" character in a replet that matches the subsequence.

Docket JP920030152US1

Appl. No.: 10/699,024

Filing Date: October 31, 2003

17. (currently amended) The method of claim 16, wherein, prior to presenting the

reconstructed represented sequence to the user, the position of the subsequence

character within the subsequence is stored in the list data structure.

18. (previously presented) The method of claim 15, wherein an indirection pointer points

to a variation so that variations common to more than one subsequence are not stored

more than once.

19. (currently amended) The method of claim 14, wherein a reconstructed represented

sequence presented to the user is in response to a query by the user.

20. (previously presented) The method of claim 19, wherein a query specifies a replet.

21. (currently amended) The method of claim 14, wherein specifying a set of one or

more initial replets comprises adding a new replet to a pre-existing set for which the

steps of claim 14 have been performed.

22. (currently amended) The method of claim 14, wherein specifying a set of one or

more initial-replets comprises removing a replet from a pre-existing set for which the

steps of claim 14 have been performed.

23. (canceled)